MAC–IME–USP

CARLOS EDUARDO FERREIRA
SALA 297A TEL.: 818 6140
E-MAIL cef@ime.usp.br

MAC 323 – Estruturas de Dados

Primeiro semestre de 1999

Exercício-Programa 2 – Entrega: **3 de maio de 1999**

Listas Ligadas

O objetivo deste exercício-programa é implementar classes com listas ligadas e listas duplamente ligadas, conforme será utilizado no projeto da disciplina MAC 211 – Laboratório de Programação. Haverá, portanto duas versões deste EP: uma para os alunos matriculados em MAC211 e outra para os que não fazem a disciplina.

1 Matriculados em MAC 211

No projeto de MAC 211 vocês estarão fazendo um pequeno sistema gerenciador de bancos de dados, o JBase. Nele a classe `BufferManager` mantém duas estruturas de dados:

- uma tabela de hash que deve ser implementada como um vetor de listas ligadas. Escreva uma classe para esta estrutura. Escreva métodos para inserir e remover elementos da tabela de hash. Faça a estrutura de forma genérica a fim de que você possa testar várias funções de hash diferentes sem muito esforço;
- uma lista duplamente ligada circular com cabeça de lista em que elementos são inseridos e removidos das duas pontas (ou seja, uma fila dupla). Faça os métodos para implementar estes operadores, e outros de listas duplamente ligadas com cabeça de lista. Escreva seus métodos de forma genérica a fim de testar várias políticas diferentes de substituição de páginas (como a LRU e MRU descritas no projeto de MAC211).

2 Não matriculados em MAC 211

Você deverá escrever as mesmas classes definidas acima, e para testá-las deverá desenvolver um pequeno sistema de escalonamento de processos de um computador com multiprocessamento.

Em um sistema de multiprocessamento a CPU executa os processos em ciclos, ou seja, nenhum job ocupa a CPU até o fim de sua execução. Após uma determinada quantidade de tempo (chamada de ciclo) o job é interrompido, o estado da memória armazenado em disco e outro job é selecionado da fila de execução para ocupar a CPU.

Existem vários métodos de escalonamento, tais como *Round Robin, shortest job first, etc.* Para você aprender mais a respeito de algoritmos de escalonamento, veja, por exemplo o livro

Modern Operating Systems, A.S. Tanenbaum, Prentice hall (1992).

Você deverá armazenar uma tabela de hash com os usuários do sistema, em que, por exemplo, você armazena a prioridade dos jobs executados pelo usuário. Os jobs que estão em execução estão armazenados em uma lista duplamente ligada com cabeça de lista, a fim de simplificar a inserção e remoção de um elemento de qualquer posição.

Para testar seu EP leia as informações dos usuários e dos jobs que devem ser executados de um arquivo. A saída do seu programa deve ser clara, a fim de que seja fácil ver qual é o job que está executando em cada instante. No final da execução mostre qual foi o tempo médio e o tempo máximo de espera dos jobs, assim como outras informações que você julgar interessante para avaliar a política de escalonamento. Assim como na outra versão, implemente diferentes estratégias de escalonamento.